Estimating the Fates of the Control Packets for Networked Control Systems with Loss of Control and Measurement Packets

Rainer Blind, Frank Allgöwer

Abstract—In this work, we extend recently published work for the estimation of fates of the control packets in Networked Control Systems (NCS). Unfortunately, the existing scheme is limited by the assumption that all measurement packets arrive. Moreover, a rank condition must hold, which is identical to a relative degree one assumption for SISO systems. In this work, we relax these assumptions and first extend the approach to NCS with measurement losses. Moreover, we present an approach to estimate the fates of the control packets for SISO systems with an arbitrary relative degree.

I. INTRODUCTION

Motivated by recent progress in microcontroller and network technology, more and more control loops are closed by a packet based digital network. Unfortunately, this approach brings a fundamental problem: the delay or loss of packets during data transfer. Consequently, control and estimation over unreliable networks has become a very interesting and active research area.

In this work, we concentrate on the estimation problem over a lossy network and ignore the possible delay of packets. Fig. 1, which is discussed in more detail in Sec. II, depicts the Networked Control System (NCS) considered in this work. Since the loop is closed over a digital network there are measurement and control packets on their way. Obviously, the estimator will suffer from measurement losses, see e.g. [1]. However, the estimator also suffers from the possible loss of control packets due to the fact that the fate of each control packet is unknown. Consequently, the estimator does not know the input to the plant. Hence, there exist different schemes in the literature to estimate the state under the constraint that control packets can get lost.

In [2], the actuator sends a “control packet delivery indicator” back to the controller. The loss of the different packets is then modeled as Jump Linear System. An acknowledgement mechanism is used to inform the estimator of the fates of the control packets in [3]–[7]. In these works, it is assumed that an acknowledgement is never lost. Moreover, it is shown how important the acknowledgement mechanism is: the separation principle only holds if there are acknowledgements. In [8], [9], it was pointed out that this assumption is unrealistic and the case where the acknowledgements can get lost is considered. It is shown that the optimal control law is in general nonlinear but there exist some special cases, where it is linear.

An UDP-like protocol (a protocol without acknowledgements) is used in [4]–[7]. It is shown that the optimal control law is in general nonlinear. Nevertheless, the problem is solved for some special cases. In [10] a suboptimal controller is designed for a NCS with an UDP-like protocol.

As already noted, the separation principle only holds if the fates of the control packets are known. Hence, their knowledge is very valuable for the controller design and it might be beneficial to estimate the fates of the control packets, as done in [11] and [12]. Unfortunately, the proposed scheme is limited by the assumption that there is no loss of measurement packets. Moreover, a rank condition must hold, which is identical to a relative degree one assumption for SISO systems.

In this work, we relax these assumptions and first extend the approach to NCS with measurement losses. Moreover, we present an approach to estimate the fates of the control packets for SISO systems with an arbitrary relative degree.

Fig. 1. The Networked Control System considered in this work.

Note that even if the actuator sends acknowledgements back to the controller, an estimation of the fate of the control packet is useful if an acknowledgement packet is lost.

The remainder of this work is as follows. First, the problem set up is given in Sec. II. Then, a scheme to estimate the fates of the control packets for a NCS with possible measurement losses is given in Sec. III. Sec. IV contains a similar scheme to estimate the fates of the control packets for SISO systems with an arbitrary relative degree. An example for each case is then given in Sec. V. Finally, Sec. VI concludes the work and gives a short outlook.
II. PROBLEM SET UP

Fig. 1 shows the model of the NCS considered in this work. It is built by the plant, the network, and the controller. A relatively simple network model is used; the fates of the measurement and control packets are modeled by the random processes $\gamma_k \in \{0,1\}$ and $\beta_k \in \{0,1\}$, respectively, where 0 indicates a loss and 1 an arrival. Moreover, we assume a linear plant of the following form:

$$
x_{k+1} = Ax_k + \beta_k Bu_k + w_k, \quad (1)$$
$$
y_k = \begin{cases} 
Cx_k + v_k & \text{if } \gamma_k = 1, \\
\emptyset & \text{if } \gamma_k = 0, 
\end{cases} \quad (2)
$$

where $x_k \in \mathbb{R}^n$ is the state vector, $u_k \in \mathbb{R}^r$ is the control input, and $y_k \in \mathbb{R}^m$ the measurement. Moreover, $w_k \in \mathbb{R}^n$ and $v_k \in \mathbb{R}^m$ are the process and measurement noise, respectively. Looking at Eq. (1), we see that the system runs open-loop if a control packet is lost. By Eq. (2), we point out that nothing arrives if a measurement is lost. Note that in other publications the arrival of pure noise is assumed in this case, see e.g. [1].

Finally, we do not make any assumptions about the control law but assume that there is a controller and two estimators: the state estimator, which estimates the state of the plant and the fate estimator, which estimates the fates of the control packets. In order to estimate the fates of the control packets, we require that the control $u_k$ is known to the fate estimator.

Moreover, we assume a state estimator of the following form:

$$
\dot{x}_{k+1} = A\hat{x}_k + \hat{\beta}_k Bu_k + \gamma_k L_k (y_{k+1} - CA\hat{x}_k - \hat{\beta}_k CBu_k), \quad (3)
$$

where $\hat{\beta}_k \in \{0,1\}$ is the estimate of the fate of the control packet. Obviously, the correction is only done if a measurement arrives.

Note, that we can extend $y_{k+1}$ by starting with $y_{k+1} = Cx_{k+1} + v_{k+1} = CAx_k + \beta_k CBu_k + Cw_k + v_{k+1}$ and continue $\xi$ times to get:

$$
y_{k+1} = CA^{\xi+1} x_{k-\xi} + \sum_{i=0}^{\xi} \beta_{k-i} CA^i Bu_{k-i} + \eta_{k-\xi,k}, \quad (4)
$$

where

$$
\eta_{k_1,k_2} := \sum_{i=0}^{k_2-k_1} CA^i w_{k_2-i} + v_{k_2+1}. \quad (5)
$$

General fate estimation scheme: The fates of the control packets will be estimated by minimizing $\|y_{k+1} - \hat{y}_{k+1}\|^2$, where $\hat{y}_{k+1} = C\dot{x}_{k+1}$, over the possible fates of the control packets. Depending on the presence of measurements and the relative degree, we extend $\hat{y}_{k+1}$ similar to Eq. (4), to get suitable equations for the fate estimation. For simplicity of notation, we define the error of the fate estimation as $\zeta_k := \beta_k - \hat{\beta}_k$, $\zeta_k \in \{-1,0,1\}$.

We can ask when the estimate of the fate of the control packet is correct. This question has been answered in [11] and [12] under the assumption that all measurements arrive and that $\text{rank}(CB) = \text{rank}(B)$ holds. In the present paper, we first extend [11] and [12] to NCS with measurement losses and finally to SISO systems with an arbitrary relative degree.

III. LOSS OF MEASUREMENTS

In general, the fate of the control packet is estimated at each time step. Clearly, this is only possible if the measurement arrives. If a measurement is lost, then the next measurement must be used to estimate the fates of the current and the next control packet.

In this section, we describe how to estimate the fates of the control packets if measurements can get lost. The basic idea is best understood if we assume that after each lost measurement a measurement arrives. Hence, we restrict the discussion to maximally one consecutive loss in the following subsection. The results for the general case are then straightforward and given in Sec. III-B.

For simplicity, we assume that the effect of the input is always present at the output, i.e. $\text{rank}(CA^i B) = \text{rank}(B)$, $\forall i \in \{0,1,\ldots\}$ throughout this section. For SISO systems, it will then be relatively straightforward to combine the results of this section with the ones of Section IV.

A. One Loss

If the previous measurement is lost, then the fate of the previous control packet could not be estimated. Consequently, it has to be estimated at the current time step, too. Therefore, we extend $\hat{y}_{k+1}$ by $\hat{y}_{k+1} = CA^2 \hat{x}_{k+1} + \hat{\beta}_{k-1} CA Bu_{k-1} + \hat{\beta}_k CBu_k$ and then the fate estimator minimizes $\|y_{k+1} - CA^2 \hat{x}_{k+1} - \hat{\beta}_{k-1} CA Bu_{k-1} - \hat{\beta}_k CBu_k\|^2$ over $\hat{\beta}_{k-1}$ and $\hat{\beta}_k$. A sufficient condition for correct fate estimations is:

Theorem 1: Suppose, we minimize $\|y_{k+1} - CA^2 \hat{x}_{k+1} - \hat{\beta}_{k-1} CA Bu_{k-1} - \hat{\beta}_k CBu_k\|^2$ over $\hat{\beta}_{k-1}$ and $\hat{\beta}_k \in \{0,1\}$ to get an estimate of the fates of the control packets. If

$$
u_k^TB^TC^TBu_k > 2|u_k^TB^TC^T(CA^2e_{k+1-1} + \eta_{k-1,k})|, \quad (6)$$

and

$$
u_{k-1}^TB^TA^TC^TCABu_{k-1} > 2|u_{k-1}^TB^TA^TC^T(CA^2e_{k-1} + \eta_{k-1,k})|, \quad (7)
$$

and

$$
2(\xi_{k-1} CA Bu_{k-1} + \xi_k CBu_k)^T(CA^2e_{k+1-1} + \eta_{k-1,k})
+ (\xi_{k-1} CA Bu_{k-1} + \xi_k CBu_k)^2 > 0 \quad \forall \xi_{k-1}, \xi_k \in \{-1,1\}
$$

(8)

where $\eta_{k-1,k}$ is given by Eq. (5), then the estimate of the fates of the control packets is correct, i.e., $\hat{\zeta}_{k-1} = \hat{\zeta}_k = 0$.

The proof, which is an extension of the one in [11] and [12], is as follows:
Proof:
\[
\|y_{k+1} - CA^2 \hat{x}_{k-1} - \hat{\beta}_{k-1}CABu_{k-1} - \hat{\beta}_kCBu_k\|^2
= \|CA^2x_{k-1} + \eta_{k-1,k} + \beta_{k-1}CABu_{k-1} + \beta_kCBu_k
- CA^2\hat{x}_{k-1} - \hat{\beta}_{k-1}CABu_{k-1} - \hat{\beta}_kCBu_k\|^2
= \|CA^2x_{k-1} + \eta_{k-1,k} + (\beta_{k-1} - \hat{\beta}_{k-1})CABu_{k-1}
+ (\beta_k - \hat{\beta}_k)CBu_k\|^2
\]
\[
= (CA^2e_{k-1} + \eta_{k-1,k})^T(CA^2e_{k-1} + \eta_{k-1,k})
+ 2((\beta_{k-1} - \hat{\beta}_{k-1})CABu_{k-1} + (\beta_k - \hat{\beta}_k)CBu_k)^T
\times (CA^2e_{k-1} + \eta_{k-1,k})
+ ((\beta_{k-1} - \hat{\beta}_{k-1})CABu_{k-1} + (\beta_k - \hat{\beta}_k)CBu_k)^T
\times ((\beta_{k-1} - \hat{\beta}_{k-1})CABu_{k-1} + (\beta_k - \hat{\beta}_k)CBu_k).
\]
First note that \((CA^2e_{k-1} + \eta_{k-1,k})^T(CA^2e_{k-1} + \eta_{k-1,k}) \geq 0\) and is independent of \(\hat{\beta}_{k-1}\) and \(\hat{\beta}_k\). Hence, we can remove it from minimization. Then, we use the shortcuts \(\zeta_k\) and \(\zeta_{k-1}\) instead of \(\beta_{k-1} - \hat{\beta}_{k-1}\) and \(\beta_k - \hat{\beta}_k\), to get:
\[
J(\zeta_{k-1}, \zeta_k)
= 2(\zeta_{k-1}CABu_{k-1} + \zeta_kCBu_k)^T(CA^2e_{k-1} + \eta_{k-1,k})
+ (\zeta_{k-1}CABu_{k-1} + \zeta_kCBu_k)^T(\zeta_{k-1}CABu_{k-1} + \zeta_kCBu_k).
\]
Obviously, \(J(\zeta_{k-1}, \zeta_k) = 0\) if \(\zeta_{k-1} = 0\) and \(\zeta_k = 0\), i.e., if the estimates of the control packets are correctly estimated. Since \(J(\zeta_{k-1}, \zeta_k)\) is minimized to estimate the fates of the control packets, we require \(J(\zeta_{k-1}, \zeta_k) > 0\) if the estimation is not correct. Therefore, we distinguish the following three cases:

- \(\zeta_{k-1} = 0, \zeta_k = \pm 1\): Here, we have:
  \[
  J(0, \pm 1) = \pm 2u_k^T(B^T)^T(CA^2e_{k-1} + \eta_{k-1,k})
  + u_k^TB^T(CA^2e_{k-1} + \eta_{k-1,k}) > 0,
  \]
  which holds if Eq. (6) of Thm. 1 holds.

- \(\zeta_{k-1} = \pm 1, \zeta_k = 0\): Here, we have:
  \[
  J(\pm 1, 0) = \pm 2u_k^T(B^T)^TCT(CA^2e_{k-1} + \eta_{k-1,k})
  + u_k^T(B^T)^TCTCABu_{k-1} > 0,
  \]
  which holds if Eq. (7) of Thm. 1 holds.

- \(\zeta_{k-1} = \pm 1, \zeta_k = \pm 1\): Here, we have:
  \[
  J(\pm 1, \pm 1) = 2(\pm CABu_{k-1} \pm CBu_k)^T(CA^2e_{k-1} + \eta_{k-1,k})
  + (\pm CABu_{k-1} \pm CBu_k)^T(\pm CABu_{k-1} \pm CBu_k) > 0,
  \]
  which holds if Eq. (8) of Thm. 1 holds.

If we consider a SISO system, then this theorem can be simplified to the following corollary:

**Corollary 1: Given a SISO system. Suppose, we minimize**
\[
\|y_{k+1} - CA^2\hat{x}_{k-1} - \hat{\beta}_{k-1}CABu_{k-1} - \hat{\beta}_kCBu_k\|^2
\]
\text{over }\hat{\beta}_{k-1} \text{ and } \hat{\beta}_k \in \{0, 1\} \text{ to get an estimate of the fates of the control packets. If all the following three equations hold:}

\[
|CBu_k| > 2|CA^2e_{k-1} + \eta_{k-1,k}|,
\]
\[
|CBu_{k-1}| > 2|CA^2e_{k-1} + \eta_{k-1,k}|,
\]
\[
|CBu_{k-1}| - |CBu_k| > 2|CA^2e_{k-1} + \eta_{k-1,k}|,
\]
\text{with }\eta_{k-1,k} \text{ as in Eq. (5), then the estimate of the fates of the control packets is correct, i.e., } \hat{\beta}_{k-1} = \beta_{k-1} = 0.

**Remark 1:** Since \(C\) and \(AB\) are scalars here, we conclude that \(|u_{k-1}|\) and \(|u_k|\) must be large enough to guarantee a correct fate estimation. By Eq. (11), we see that the difference between \(|CBu_{k-1}|\) and \(|CBu_k|\) must be large enough, too.

**B. Arbitrary number of consecutive losses**

First, we define \(k^*\) as the time instance where the last measurement has arrived. Note, that \(k^* = k\) if there was no measurement loss. The fate estimator finds \(\hat{\beta}_{k^*}, \ldots, \hat{\beta}_{k}\) by minimizing \(\|y_{k+1} - \hat{y}_{k+1}\|^2\), where \(\hat{y}_{k+1}\) is extended up to the time where the last measurement has arrived:
\[
\arg \min_{\hat{\beta}_{k^*}, \ldots, \hat{\beta}_k} \|y_{k+1} - CA^{k^* + k^* - \hat{\beta}_{k^*}, k^* + k^* - \hat{\beta}_k} \hat{x}_{k^*} - \sum_{i=0}^{k-k^*} \hat{\beta}_{k-i}CA^IBu_{k-i}\|^2.
\]
\text{over }\hat{\beta}_{k^*}, \ldots, \hat{\beta}_k \in \{0, 1\}

(12)

The following theorem gives a sufficient condition for a correct fate estimation.

**Theorem 2:** Suppose, Eq. (12) is used to estimate the fates of the control packets. If for all \(\xi_{k^*}, \ldots, \xi_k \in \{-1, 0, 1\}, \text{ except for } \xi_{k^*} = 0, \ldots, \xi_k = 0:\)
\[
2 \Xi_{k^*, k}(CA^{k^* + k^* - \hat{\beta}_{k^*}, k^* + \eta_{k^*}} + \Xi_{k^*, k} \Xi_{k^*, k} > 0,
\]
\text{where}
\[
\Xi_{k^*, k} = \sum_{i=0}^{k-k^*} \xi_{k-i}CA^IBu_{k-i}
\]
\text{and }\eta_{k^*} \text{ is given in Eq. (5), then the estimation of the fates of the control packets is correct.}

The proof follows the same lines as the one of Thm. 1 and is thus omitted.

**Remark 2:** It is possible to guarantee that Eq. (13) is always fulfilled if we assume a maximal number of consecutive losses and choose \(u_k\) large enough, see Sec. III-C for details.

**Remark 3:** The fate estimations are only calculated if a measurement packet arrives. Hence, if it is lost, we have to choose \(\hat{\beta}_k\) differently. E.g., \(\hat{\beta}_k = 1\) or \(\hat{\beta}_k = E[\beta]\). However, when the next measurement arrives, \(\hat{\beta}_k\) will be recalculated.

Again, we can simplify this theorem for SISO systems to get the following corollary:

**Corollary 2:** Given a SISO system. Suppose, we minimize
\[
\|y_{k+1} - CA^{k^* + k^* - \hat{\beta}_{k^*}} \hat{x}_{k^*} - \sum_{i=0}^{k^* - k^*} \hat{\beta}_{k-i}CA^IBu_{k-i}\|^2
\]
\text{over }\hat{\beta}_{k^*}, \ldots, \hat{\beta}_k \in \{0, 1\} \text{ to get an estimate of the fates of the packets. If for all }\xi_{k^*}, \ldots, \xi_k \in \{-1, 0, 1\}, \text{ except for } \xi_{k^*} = 0, \ldots, \xi_k = 0:
\[
\min_{\xi_{k-1}} \sum_{i=0}^{k-k^*} \xi_{k-i}CA^IBu_{k-i} > 2CA^{k^* + k^* - \hat{\beta}_{k^*}, k^* + \eta_{k^*}}.
\]

where \( \eta_{k^*,k} \) is given in Eq. (5), then the estimation of the fates of the control packets is correct.

C. Bounds for \( e_{k^*} \) and \( \eta_{k^*,k} \) to guarantee a correct fate estimation

Unfortunately, \( e_{k^*} \) and \( \eta_{k^*,k} \) are not known exactly and hence we cannot check the conditions of Thm. 1 or Thm. 2 directly. However, as in [11] and [12] we can bound \( e_{k^*} \) and \( \eta_{k^*,k} \) if the initial conditions and the noise are norm bounded, i.e., if

\[
\|x_0\| \leq \delta_x, \quad \|e_0\| \leq \delta_e, \quad \|w_k\| \leq \delta_w, \quad \|v_k\| \leq \delta_v.
\]

With this, \( \eta_{k^*,k} \) can be easily bounded:

\[
\|\eta_{k^*,k}\| \leq \sum_{i=0}^{k-k^*} \|CA^i\|\delta_w + \delta_v.
\]

Moreover, if we assume that the number of consecutive losses is bounded by \( l \), then we have:

\[
\|\eta_{k^*,k}\| \leq \|\eta_{k-l,k}\| \leq \sum_{i=0}^{l} \|CA^i\|\delta_w + \delta_v.
\]

The estimation error \( e_{k^*} \) is

\[
e_{k^*} = \prod_{j=0}^{k^*-1}(A-\gamma_jLCA)e_0 + \sum_{j=0}^{k^*-1} \prod_{i=j}^{k-1} (A-\gamma_iLCA)h_j
\]

with \( h_j = c_j(B-\gamma_jLCB)u_j + z_j, \quad z_j = w_j - \gamma_jL(Cw_j + \gamma_{j+1}). \) Note, that for simplicity of notation, we assume a constant observer gain \( L \) here. If we assume that the fates of the control packets are estimated correctly up to time \( k^*-1 \), then the state estimation error at time \( k^* \) can be bounded as follows:

\[
\|e_{k^*}\| \leq \prod_{j=0}^{k^*-1}(A-\gamma_jLCA)\|\delta_z + \sum_{j=0}^{k^*-1} \prod_{i=j}^{k-1} (A-\gamma_iLCA)\|\delta_z,
\]

where \( \delta_z = \delta_w + \|LC\|\delta_w + \|L\|\delta_v. \)

Now, we can use these bounds for \( \|\eta_{k^*,k}\| \) and \( \|e_{k^*}\| \) to guarantee that the fates of the control packets are estimated correctly from time \( k^* \) up to \( k \). Therefore, we have to restrict the input \( u \) such that the conditions of Thm. 2 hold.

At first glance, this might be possible by choosing just \( u_k \) correctly. Unfortunately, Eq. (13) must hold even if \( \xi_k = 0 \) but \( \xi_{k-1}, \ldots, \xi_{k^*} \neq 0 \), i.e., without the influence of \( u_k \). This problem can be solved in a recursive manner. Assuming \( \xi_{k-1} \neq 0 \), we have to choose \( u_{k-1} \) at time \( k-1 \) such that Eq. (13) is fulfilled for the case that \( \xi_k = 0 \). Consequently, at each time step \( k \), \( k^* \leq k < k \) we have to choose \( u_k \) such that also some future equations hold. This is only possible if a maximal number of consecutive losses is assumed.

Since the worst case is used throughout, the bounds for \( e_{k^*} \) and \( \eta_{k^*,k} \) will be very conservative in general; possibly too conservative for practical applications. This means that very large values of \( u_k \) are necessary to guarantee a correct fate estimation. Obviously, this is not always satisfying. However, in the examples of Sec. V, we see that we get good results even if the input \( u \) is not restricted to satisfy the conditions of Thm. 2. From an other point of view, Thm. 2 tells us that the fate estimates are not estimated correctly if the input \( u \) is relatively small. In this case, there will be no dramatic consequences if the fates of the control packets are wrong.

To sum up, the estimates of the fates of the control packets are found by minimizing \( \|y_{k+1} - \hat{y}_{k+1}\|^2 \) over the possible fates of the control packets. If measurements are lost, then \( \hat{y}_{k+1} \) has to be extended up to the time where the last measurement arrived. Finally, if we assume a maximal number of consecutive losses, then the control \( u \) can be restricted such that the fate estimations are guaranteed to be correct.

IV. ARBITRARY RELATIVE DEGREE

In this section, we show how the fates of the control packets can be estimated for a SISO system with relative degree \( r > 1 \). For simplicity, we assume that all measurements arrive, i.e., \( \gamma_k = 1 \) throughout this section.

Remember the definition of the relative degree \( r \): \( CA^{-1}B = 0 \) for all \( i < r \) and \( CA^{-1}B \neq 0 \). Since the input does not appear immediately at the output if \( r > 1 \), the fate of the current control packet can not be estimated; only the fate of the control packet \( r-1 \) time steps ago can be estimated. Therefore, we extend \( y_{k+1} \) until the input appears:

\[
y_{k+1} = CA^r x_{k-r+1} + \beta_{k-r+1}CA^{-1}Bu_{k-r+1} + \eta_{k-r+1,k},
\]

and the fate estimator minimizes \( \|y_{k+1} - CA^r \hat{x}_{k-r+1} - \beta_{k-r+1}CA^{-1}Bu_{k-r+1}\|^2 \) in order to find \( \beta_{k-r+1} \). Again, the following theorem gives a sufficient condition for a correct fate estimation.

Theorem 3: Suppose, we minimize \( \|y_{k+1} - CA^r \hat{x}_{k-r+1} - \beta_{k-r+1}CA^{-1}Bu_{k-r+1}\|^2 \) over \( \beta_{k-r+1} \in \{0, 1\} \) to get an estimate of the fate of the control packet. If

\[
|CA^{-1}Bu_{k-r+1}| > 2|CA^r e_{k-r+1} + \eta_{k-r+1,k},
\]

where \( \eta_{k-r+1,k} \) is given in Eq. (5), then the estimate of the fate of the control packet fate is correct, i.e., \( \xi_{k-r+1} = 0 \). The proof follows the same lines as the one of Thm. 1 and is thus omitted.

Remark 4: Again, we see that the fate estimation will be correct if \( |u_{k-r+1}| \) is large enough.

Remark 5: Note that the output \( y_{r+1} \) is used to get the fate estimate \( \beta_{k-r+1} \). There is no estimate of \( \beta_{k-r+2} \) up to \( k \), i.e., there is no estimate of the fates of the control packets for the period \( k-r+2 \) to \( k \). During this period another value must be used for \( \beta_{k} \), e.g. its expected value \( E[\beta_{k}] \).

After \( \beta_{k-r+1} \) is found, all the estimated states \( \hat{x}_{k-r+1} \) up to \( \hat{x}_k \) must be recalculated.

V. EXAMPLES

In this section, we show the benefits of the proposed schemes using two examples.

A. Measurement Losses

First, we present an example, where the control and measurement packets can get lost. The discussion is split in two parts. First, we simulate the system with the estimators
100,000 times and show the mean values. Then we exemplary discuss the results of one particular simulation.

We consider a system given by Eq. (1) and (2) with the following matrices:

$$A = \begin{bmatrix} 1.4 & 0.1 \\ 0.3 & 1.2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$  

The eigenvalues of $A$ are 1.5 and 1.1 and hence the system is unstable. In order to stabilize the system, we use a state feedback control law $u_k = F\hat{x}_k$, where $F$ was chosen such that the closed loop eigenvalues are 0.6 and 0.7. For each simulation, the initial conditions $x_0$ and $\hat{x}_0$ are chosen randomly such that: $\|x_0\| \leq 100$ and $\|e_0\| \leq 20$. Moreover, we assume that 95% of the packets arrive, i.e., we have $E[\beta_k] = 0.95$ and $E[\gamma_k] = 0.95$. Note that we get an arbitrary number of consecutive losses here. Finally, the process and measurement noise is Gaussian white noise with zero mean and covariance matrices $Q = I$ and $R = 1$, respectively.

For the state estimation a Kalman filter is used, where the correction step is left out if the measurement is missing. To estimate the fates of the control packets, we use the scheme presented in Sec. III. When there is a measurement loss, we set $\hat{\beta}_k$ temporarily to one (see Rem. 3).

Fig. 2 shows the mean of the normed state $\|x\|$, the mean of the normed estimation error $\|\hat{e}\|$, the mean of the normed state under the assumption that $\beta_k$ is known by the estimator $\|\hat{\beta}_k\|$, and the mean of the normed input $\|u\|$. Note that during these simulations, 12.15% of the fate estimations are not correct. However, this only happens if $\|u\|$ is relatively small and hence these errors seem to be not dramatic.

In the beginning, the input $\|u\|$ is relatively large and hence Thm. 2 states that the fate estimations are correct. Thus $\|x\|$ and $\|\hat{x}\|$ coincide here. After some time, the state $\|x\|$ is coming closer to the origin and thus the input $\|u\|$ gets smaller. As a consequence thereof, the fate estimations are not always correct and hence $\|x\|$ increases.

One might wonder, whether more naive schemes for the fate estimation also work. Fig. 3 shows the mean of the normed state $\|x\|$ and the normed estimation error $\|\hat{e}\|$ of 100,000 simulation runs of the same setup except that two naive fate estimation schemes are used. The upper subfigure shows the simulations, where the very simple scheme $\beta_k = 1$ was chosen. Although, only 5% of the fate estimations are wrong, the simulations show a much worse performance than the simulations presented in Fig. 2, where the fate estimation scheme of Sec. III is used. The lower subfigure shows the simulations, where $\hat{\beta}_k = E[\beta_k]$ is used. Again, these simulations are much worse than the ones of Fig. 2.

Now, after we have seen that the mean of 100,000 simulations looks quite good if the scheme of Sec. III is used to estimate the fate of the control packets, we will have a look at the details. Fig. 4 shows the details of one particular
simulation. In the upper subfigure, the normed state, the estimation error and the control is plotted. The subfigure in the middle shows the fates of the control and measurement packets, i.e. $\beta$ and $\gamma$, respectively. Finally, in the lowest subfigure shows the fate estimation error, i.e., $\beta - \hat{\beta}$.

In the beginning and during the period $k \approx 110 \ldots 120$, we see that the input is large compared with the estimation error. Hence, the fate estimations are correct. On the other hand, during the periods $k \approx 20 \ldots 25$ and $k \approx 60 \ldots 70$, the input is small compared with the estimation error. Hence, Thm. 2 can not guarantee a correct fate estimation. This is reflected by the lowest subfigure, where we see some fate estimation errors during these intervals.

B. Relative degree $r = 2$

In order to show the results for a system with a relative degree greater than one, we change the measurement matrix of the previous example and use the following matrices:

\[
A = \begin{bmatrix} 1.4 & 0.3 \\ 0.1 & 1.2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 1 \end{bmatrix}.
\]

In order to stabilize the system, we use a state feedback control law $u_k = F \hat{x}_k$, where $F$ was chosen such that the closed loop eigenvalues are 0.8 and 0.9. We use the same bounds for the initial conditions and the same covariance matrices for the process and measurement noise as in the previous example. However, we increase the loss of control packets in this example, and assume that only 90% arrive, i.e. $E[\beta_k] = 0.9$. Finally, we assume that all measurements arrive.

Again, we use the modified Kalman filter to estimate the state and set $\hat{\beta}$ temporarily to one (see Rem. 5). To estimate the fates of the control packets, we use the scheme presented in Sec. IV. Fig. 5 shows the mean of 100,000 simulation runs. Although even 29.83% of the fate estimations are not correct the simulations look quite good. Again, the fates of the control packets are estimated correctly if $u$ is large enough. A fate estimation error occurs only if $u$ is relatively small and hence these errors seem to be not dramatic.

By these examples, we see that the proposed schemes are able to estimate the fates of the control packets for a NCS with loss of measurements or a relative degree greater than one. Although the fate estimation is not always correct, the closed loop simulations show that it outperforms the naive schemes.

VI. CONCLUSIONS AND OUTLOOK

In this work, we extended the results in [11] and [12] to the case of arbitrary relative degree and possible measurement losses. We showed, that there exist conditions which guarantee a correct fate estimation. The examples of Sec. V, showed that good results are achieved even if the fates of the control packets are not always estimated correctly. As in [11] and [12], it is possible to restrict the control $u$ such that Thm. 2 and Thm. 3 always guarantees a correct fate estimation. This might be very interesting if a model predictive control (MPC) approach is used.

REFERENCES


